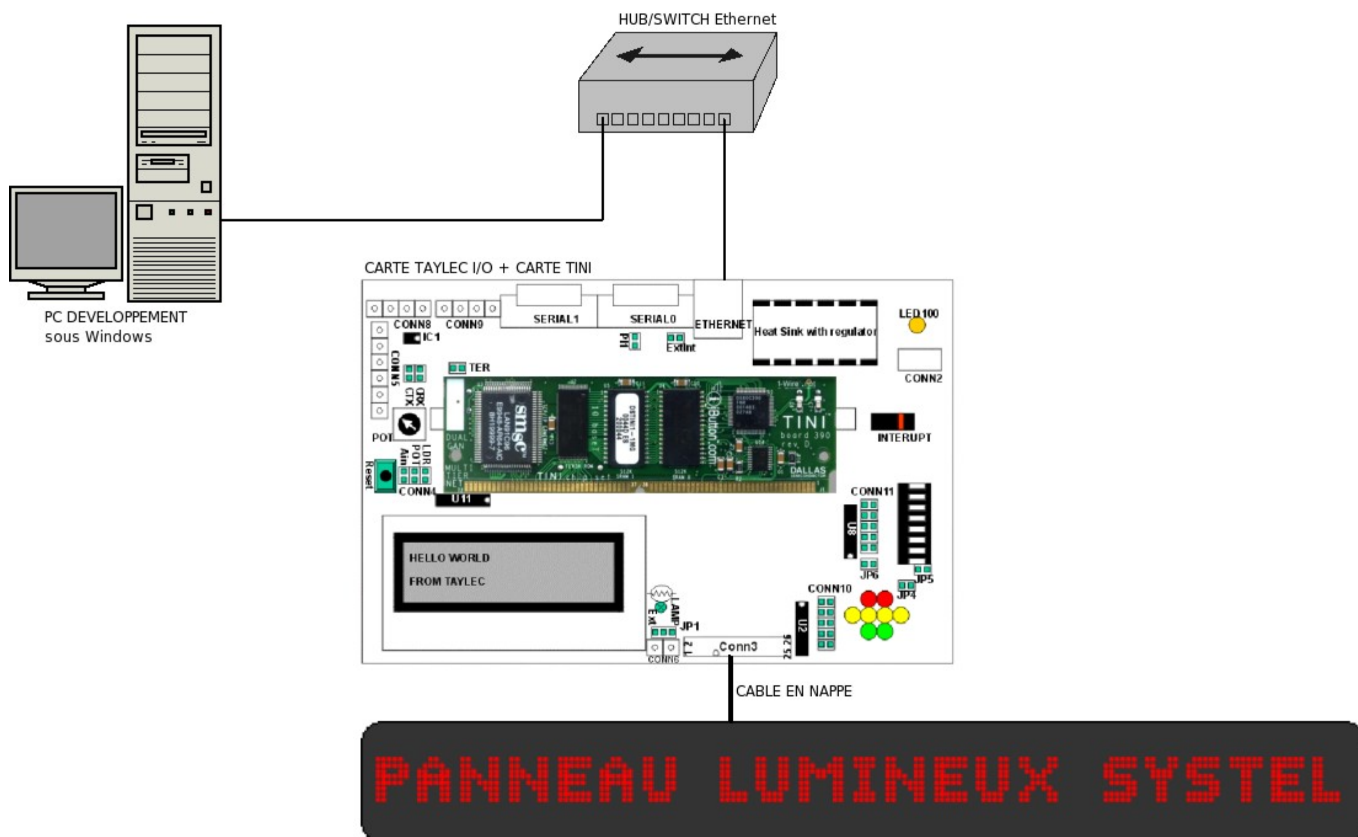
Manuel d' Utilisation

Table des matières

Remarque : il est fortement conseillé de lire les annexes sur les parties Réseau et Programmation.

Présentation

On dispose de l'environnement de développement suivant :



Introduction

Ce manuel d'utilisation de la carte TINI se décompose en plusieurs parties :

- Partie A : installation et mise en oeuvre du matériel et logiciel (→ par le technicien IRIS)
- Partie B : mise en oeuvre de la communication réseau (PC_DEV ↔ TINI)
- Partie C : fabrication d'un exécutable pour la carte TINI (→ poste de développement PC_DEV)
- Partie D : déploiement d'une application (PC_DEV → TINI)
- Partie E : exécution d'une application (PC_DEV → TINI)

Matériel nécessaire

Le matériel à votre disposition pour cette activité est le suivant :

- un PC sous Windows équipé d'une carte réseau Ethernet
- un HUB (concentrateur) ou un SWITCH (commutateur) et les câbles réseaux
- un carte TINI avec sa carte support Taylec
- un panneau lumineux de marque Systel

LOGICIEL NÉCESSAIRE

La liste des logiciels à mettre en oeuvre est la suivante :

Nom	*Description*	*Rôle*
PuTTY	PuTTY est un émulateur de terminal doublé d'un client pour les protocoles SSH, Telnet, rlogin, et TCP brut. Il permet également d'établir des connexions directes par liaison série RS-232. A l'origine disponible uniquement pour Windows, il est à présent porté sur diverses plate-formes Unix (et non-officiellement sur d'autres plate-formes). C'est un logiciel libre distribué selon les termes de la licence MIT. Lien : http://www.chiark.greenend.org.uk/~sgtatham/putty/	permet d'ouvrir, à partir d'un PC, une session Telnet sur la carte TINI et donc d'interagir avec celle-ci.
FileZilla	FileZilla est un client FTP, FTPS et SFTP, multiplate-forme depuis la version 3. C'est un logiciel libre développé sous la licence publique générale GPL/GNU. Lien : http://filezilla-project.org/	permet de transférer des fichiers du poste de développement (PC_DEV) vers la carte TINI.
Notepad++	Notepad++ est un éditeur de texte générique écrit en C++, qui intègre de nombreuses fonctionnalités dont la coloration syntaxique de code source pour beaucoup de langages (C, C++, Java, etc ..). Il est également une alternative au bloc-notes de Windows (d'où le nom). Le projet est sous licence GPL. Lien : http://notepad-plus.sourceforge.net/fr/site.htm	permet d'éditer les fichiers sources de l'application sur le poste de développement (PC_DEV).
Windows ©	Windows est une gamme de systèmes d'exploitation produite par Microsoft ©, principalement destinés aux ordinateurs compatibles PC. C'est le successeur de MS-DOS. Windows est distribuée sous la licence Microsoft EULA (End User License Agreement). La plupart des ordinateurs vendus ont un système Windows préinstallé par le constructeur sous licence OEM (Original Equipment Manufacturer). Contrairement aux autres logiciels utilisés ici, Windows fait partie des logiciels propriétaires. Lien : http://www.microsoft.com/france/windows/	permet d'exploiter le poste de développement (PC_DEV).
cmd.exe	cmd.exe est l'interpréteur de commande en mode texte de Windows. Il est l'équivalent du shell sous Unix. Il peut interpréter des fichiers batch qui sont des fichiers textes contenant une série de commandes. Un fichier batch est l'équivalent d'un script shell sous Unix. Les fichiers batch de Windows sont des fichiers dotés de l'extension .BAT (ou .CMD). Sous Windows, l'appellation "shell" regroupe deux concepts : • Comme pour Unix, l'interpréteur en ligne de commande (command.com pour les versions de Windows basées sur MS-DOS, et cmd.exe pour celles qui reposent sur Windows NT) • L'interface graphique, en général l'Explorer. *Remarque :* avec Windows Vista est apparu un nouvel interpréteur de commande, le Windows PowerShell (utilisable aussi sous Windows XP), orienté objet. Lien : http://windows.developpez.com/cours/ligne-commande/	permet d'exécuter des commandes (ping, ipconfig, ...) et d'automatiser la procédure de fabrication de l'application (maketini.bat).

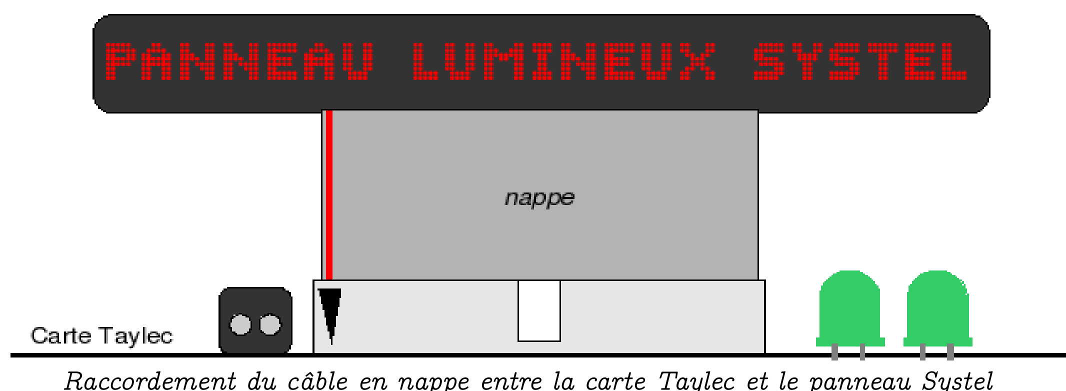
Nom	*Description*	*Rôle*
Java	Java est une technologie développée par Sun Microsystems. Elle correspond à plusieurs produits et spécifications de logiciels qui, ensemble, constituent un système pour développer et déployer des applications. Java est utilisée dans une grande variété de plates-formes depuis les systèmes embarqués et les téléphones mobiles jusqu'aux serveurs, les applications d'entreprise, les superordinateurs et dans une moindre mesure pour les interfaces graphiques comme les applets Java du Web. Java est sous licence GNU GPL depuis novembre 2006. Depuis des années, Sun Microsystems appelle Java la « technologie Java » dans son ensemble. En pratique, beaucoup de programmeurs utilisent le mot « Java » pour désigner le langage de programmation, tandis que la plate-forme d'exécution est appelée « JRE » Java Runtime Environment, environnement d'exécution Java) et le système de compilation : « JDK » (Java Development Kit) plutôt que « compilateur Java ». Lien : http://java.sun.com/	Dans ce système, la technologie JAVA est utilisée de la manière suivante : « java » : le langage de programmation de l'application s'exécutant sur la carte TINI « jre » : machine virtuelle présente sur la carte TINI permettant d'exécuter l'application « jdk » : environnement de développement installé sur PC_DEV permettant de fabriquer l'application pour la carte TINI. La version du JDK utilisée dans ce projet est la 1.4.2_19.

PARTIE A : INSTALLATION ET MISE EN OEUVRE DU MATÉRIEL ET LOGICIEL

A . 1 . Installer le matériel fourni

Vous devez réaliser dans l'ordre les actions suivantes :

- relier la carte Taylec/TINI au HUB/Switch par un câble Ethernet
- relier le panneau lumineux Systel à la carte Taylec en utilisant le câble en nappe (voir schéma)
- alimenter le panneau lumineux Systel (alimentation 220V)
- alimenter la carte Taylec/TINI (adaptateur 12V fourni)
- relier le poste de développement PC_DEV au HUB/Switch par un câble Ethernet
- démarrer le poste de développement PC_DEV sous Windows



Raccordement du câble en nappe entre la carte Taylec et le panneau Systel

A . 2 . Installer les logiciels suivants : PuTTY, FileZilla.

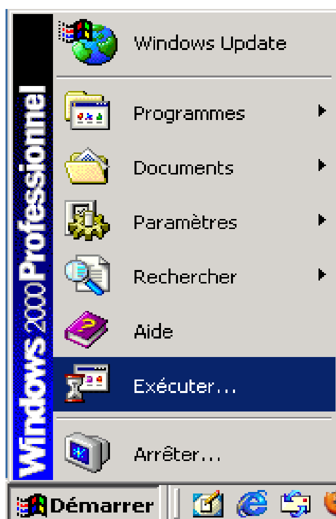
Vérifier préalablement leur présence sur le poste de développement et les installer si nécessaire (ils sont disponibles sur le serveur de la section, sur le CD ressources et par téléchargement sur Internet).

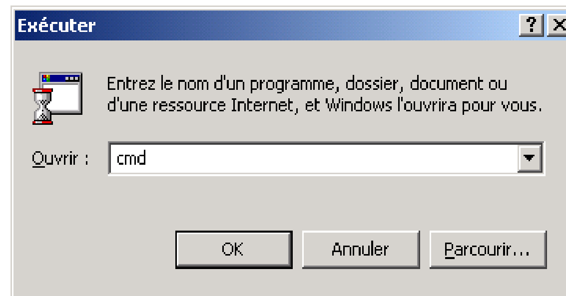
Partie B : Mise en oeuvre de la communication réseau

B . 1 . Identifier la configuration rêseau du poste de dêveloppement.

On va utiliser l'interpréteur de commandes de Windows, pour cela il faut faire :

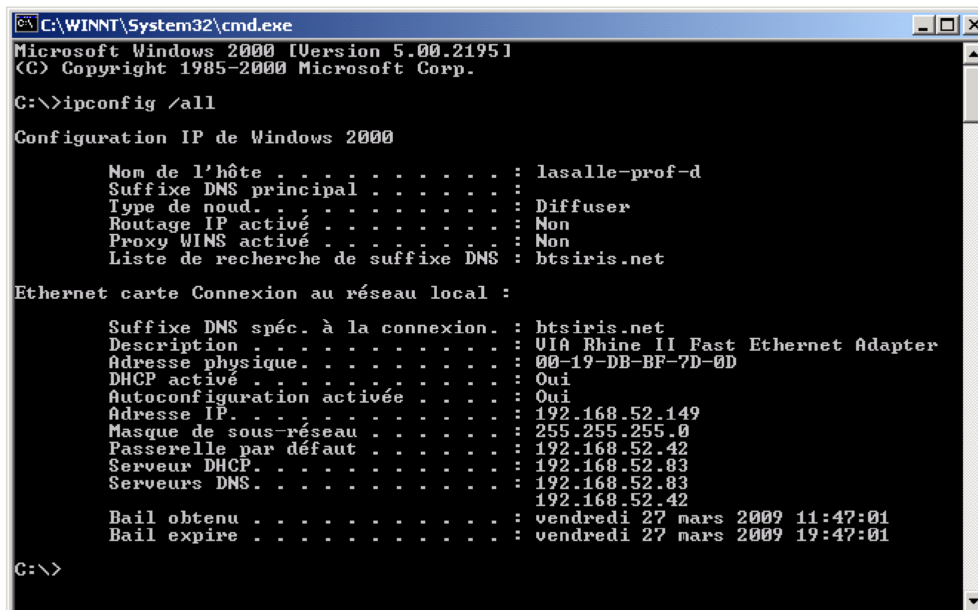
```
Menu Démarrer → Exécuter → cmd
```





Puis pour obtenir la configuration réseau du poste, taper la commande :

```
ipconfig /all
```



B . 2 . Tester la fonctionnalité réseau du poste de développement.

En utilisant l'interpréteur de commandes de Windows, taper la commande :

```
ping adresse_ip_pc
```

```
C:\WINNT\System32\cmd.exe

C:\>ping 192.168.52.149

Envoi d'une requête 'ping' sur 192.168.52.149 avec 32 octets de données :

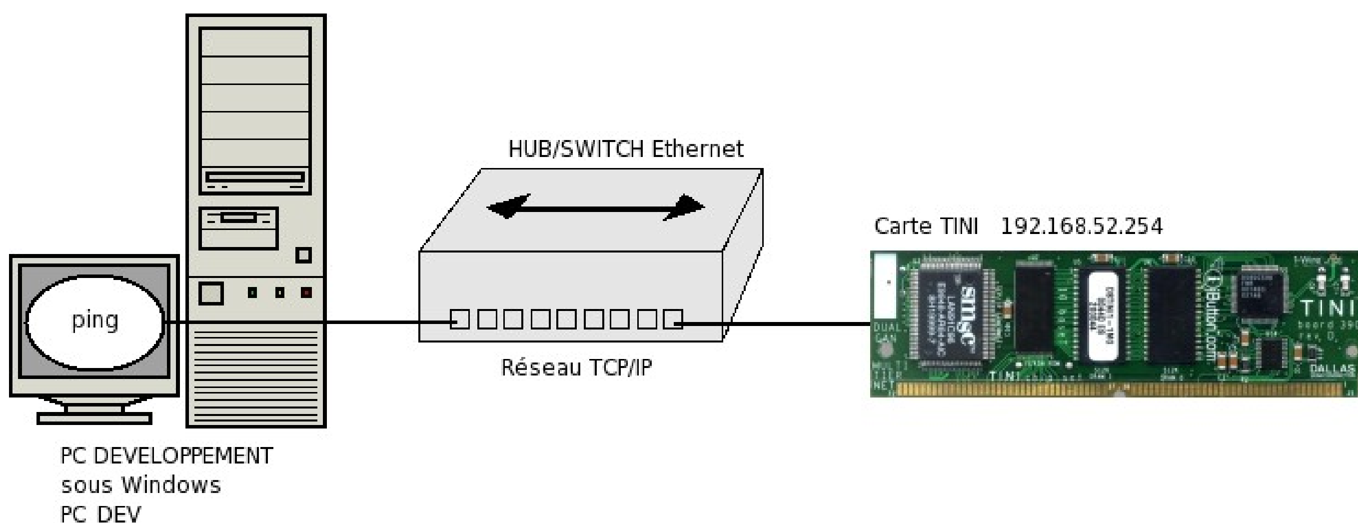
Réponse de 192.168.52.149 : octets=32 temps<10 ms TTL=128
Réponse de 192.168.52.149 : octets=32 temps<10 ms TTL=128
Réponse de 192.168.52.149 : octets=32 temps<10 ms TTL=128
Réponse de 192.168.52.149 : octets=32 temps<10 ms TTL=128

Statistiques Ping pour 192.168.52.149:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    minimum = 0ms, maximum = 0ms, moyenne = 0ms

C:\>_
```

B . 3 . Tester la liaison réseau avec le système embarqué (carte TINI).

Pour réaliser un test de base d'une liaison réseau, on utilise souvent la commande ping qui permettra, en cas de succès, de valider la pile de protocoles jusqu'au niveau IP.



En utilisant l'interpréteur de commandes de Windows, taper la commande :

```
ping adresse_ip_tini
```

A la suite de ce test, on obtient généralement une des deux situations suivantes :

Réponses aux demandes d'echo du ping :

```
C:\WINNT\System32\cmd.exe

C:\>ping 192.168.52.254

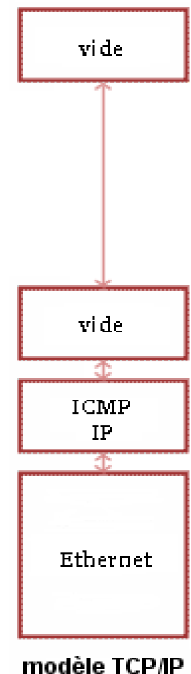
Envoi d'une requête 'ping' sur 192.168.52.254 avec 32 octets de données :

Réponse de 192.168.52.254 : octets=32 temps<10 ms TTL=255
Réponse de 192.168.52.254 : octets=32 temps<10 ms TTL=255
Réponse de 192.168.52.254 : octets=32 temps<10 ms TTL=255
Réponse de 192.168.52.254 : octets=32 temps<10 ms TTL=255

Statistiques Ping pour 192.168.52.254:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    minimum = 0ms, maximum = 0ms, moyenne = 0ms

C:\>_
```

Le succès de ce test permet de vailder la pile de protocoles jusqu'au niveau IP. Par contre, les couches supérieures (TCP/UDP, et le niveau application) ne sont pas validées car non-couvertes par ce test (ping).



modèle TCP/IP

Pas de réponses aux demandes d'echo du ping :

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\tv>ping 192.168.52.140

Envoi d'une requête 'ping' sur 192.168.52.140 avec 32 octets de données :

Délai d'attente de la demande dépassé
Délai d'attente de la demande dépassé
Délai d'attente de la demande dépassé

Statistiques Ping pour 192.168.52.140:
    Paquets : envoyés = 3, reçus = 0, perdus = 3 (perte 100%),
```

Il n'y a pas de réponse (timeout) de l'équipement d'adresse IP 192.168.52.140. Plusieurs causes possibles à ces non-réponses : pas d'équipement « en fonction » pour cette adresse, réponses aux ping filtrées, etc ...

PARTIE C : FABRICATION D'UN EXÉCUTABLE POUR LA CARTE TINI

C . 1 . Installer les ressources pour le développement

Les logiciels sont disponibles sur le serveur de la section, sur le CD ressources et par téléchargement sur Internet.

C . 1 . 1 . Installer les logiciels suivants : Notepad++ et J2SDK 1.4.2_19.

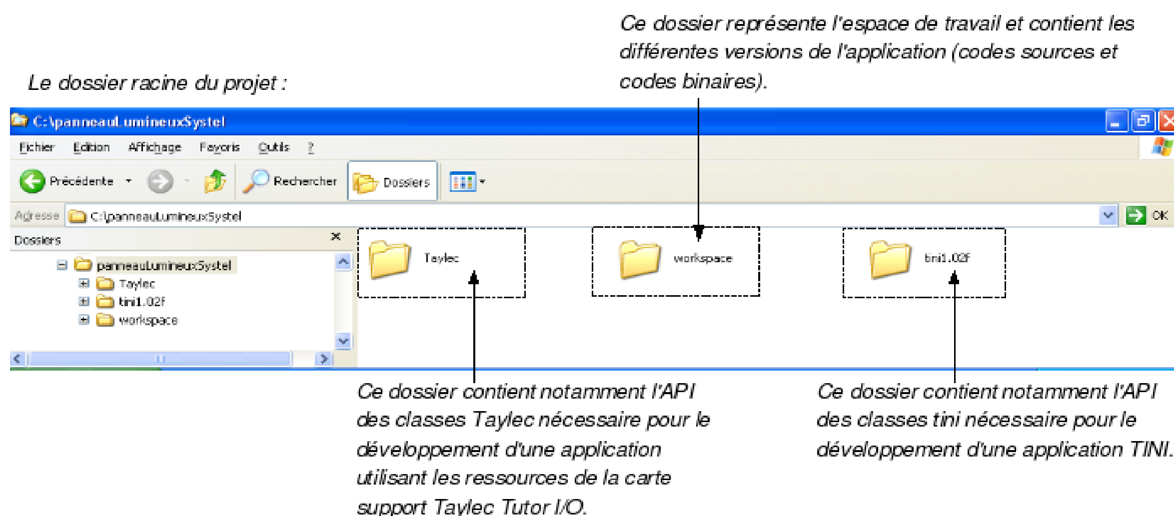
*Le JDK 1.4.2 doit être installé dans **C:**\.*

C . 1 . 2 . Créer un répertoire (dossier) racine pour le projet.

*Le répertoire (dossier) racine se nomme : **panneauLumineuxSystel** et il sera créé dans **C:**\.*

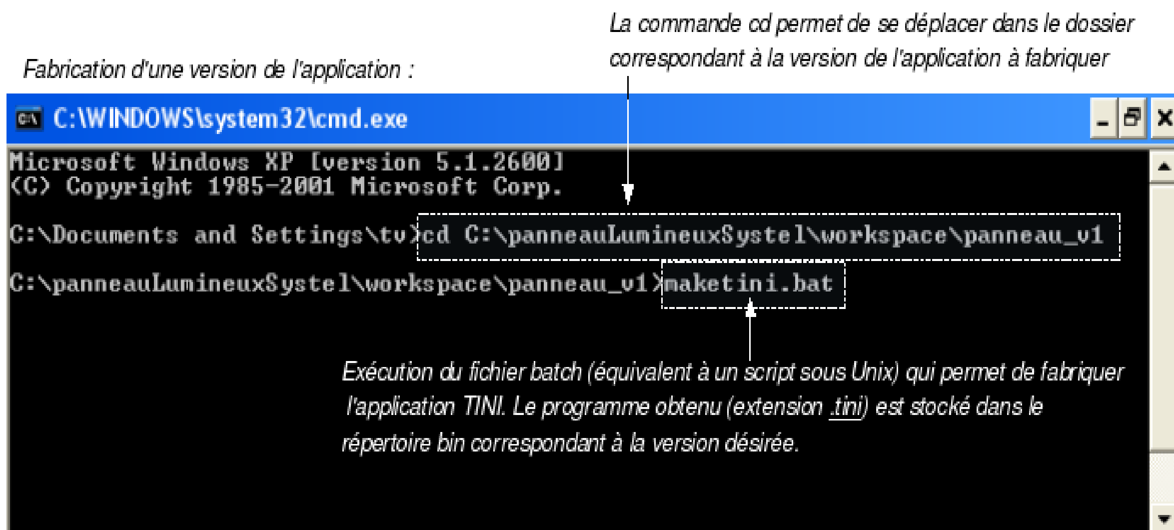
C . 1 . 2 . Installer les APIs tini1.02f et Taylec et les sources de l'application

*Pour cela, il vous faut copier les APIs tini1.02f et Taylec et les sources dans le répertoire racine **C:\panneauLumineuxSystel**.*



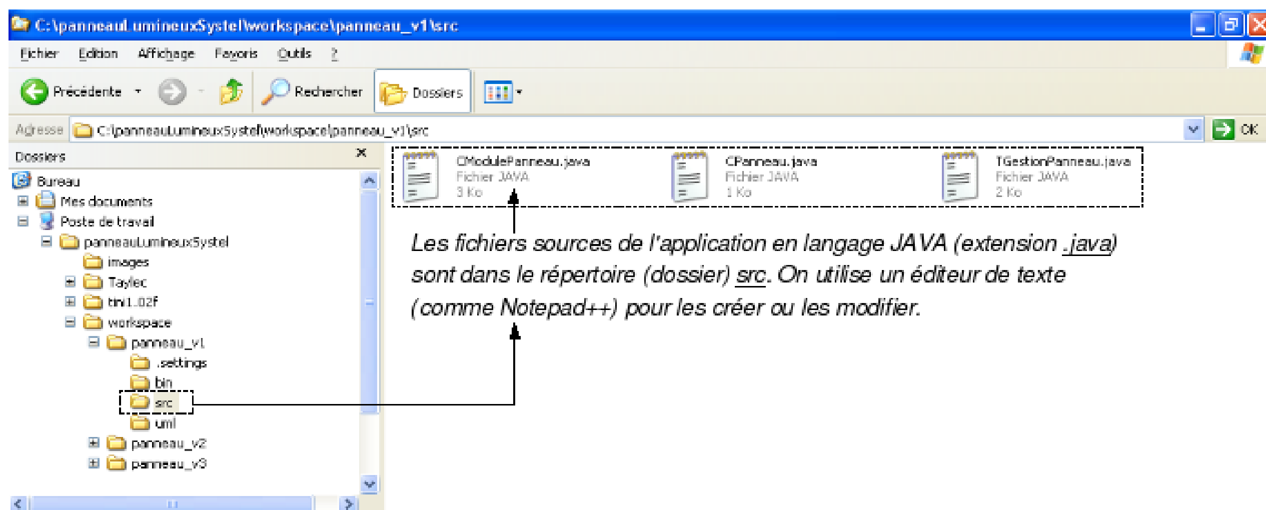
C . 2 . Fabriquer une version de l'application pour la carte TINI.

On utilise le fichier **maketini.bat** qui permet d'automatiser le cycle de fabrication d'un programme pour la carte TINI.

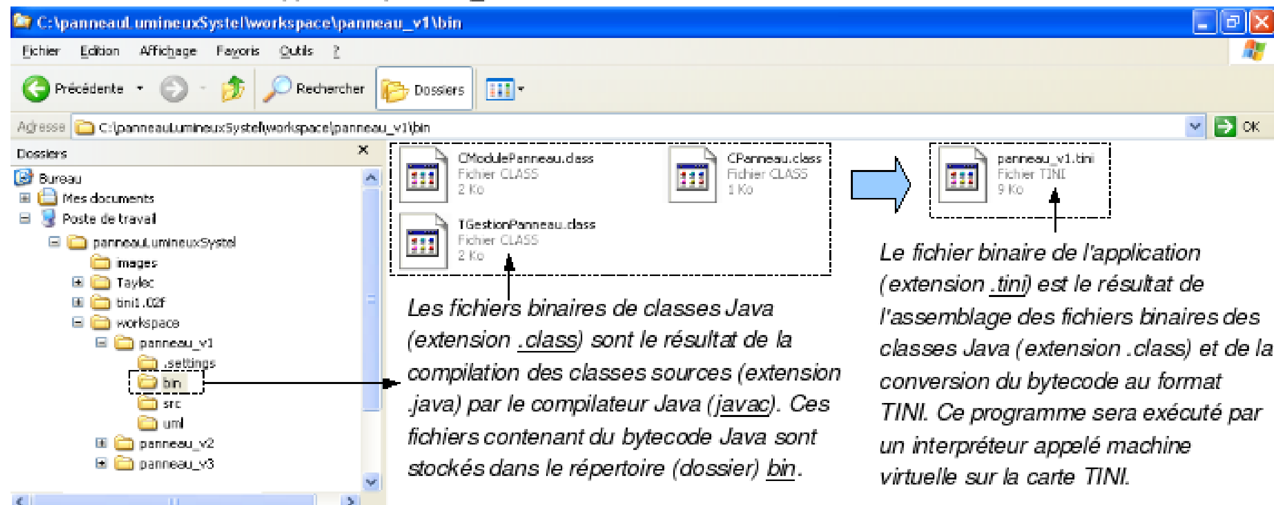


L'espace de travail (dossier **C:\panneauLumineuxSystel\workspace**) doit être le suivant :

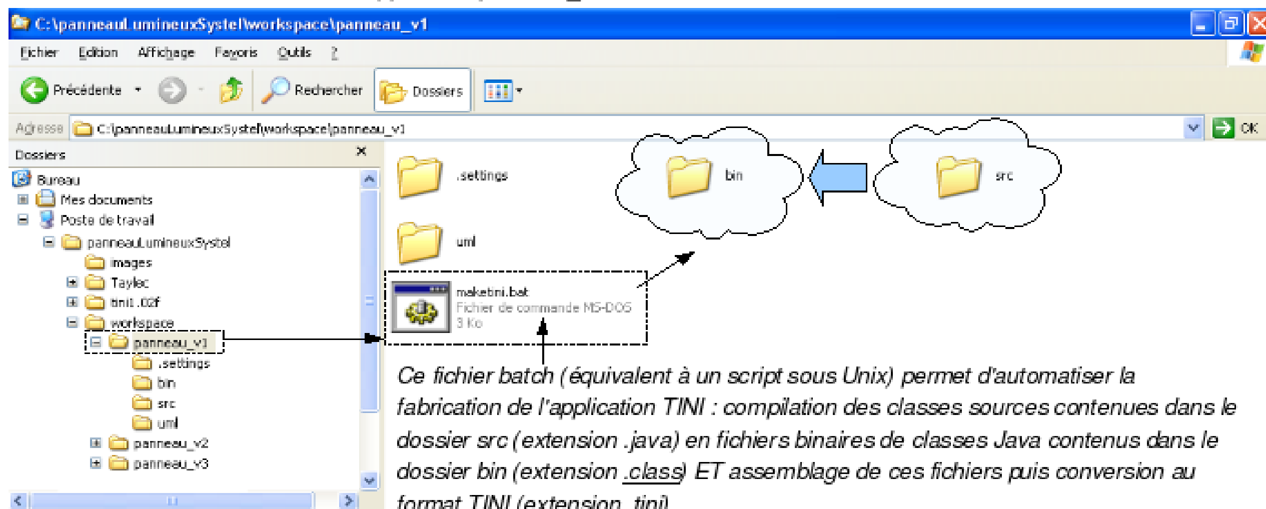
Les fichiers sources de l'application panneau_v1 :



Les fichiers binaires de l'application panneau_v1 :



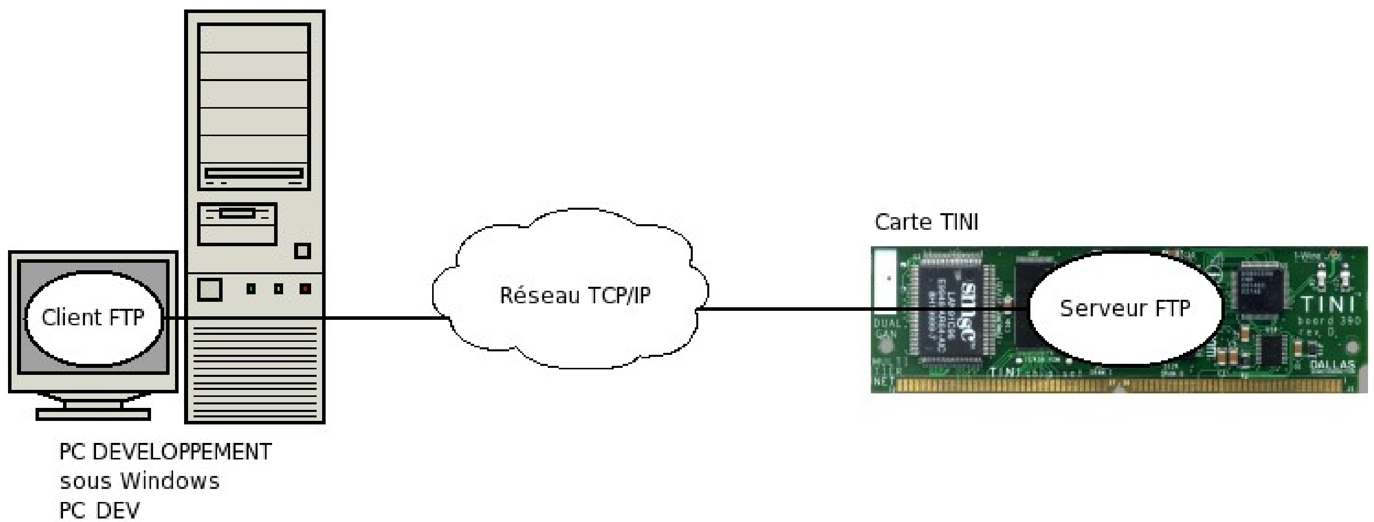
Le fichier batch de fabrication de l'application panneau_v1 :



Partie D : Déploiement d'une application

La carte TINI dispose d'un serveur FTP natif. On va donc utiliser un client FTP (comme FileZilla) pour transférer un programme **.tini** vers la carte TINI.

FTP (*File Transfer Protocol*, protocole de transfert de fichiers) est un protocole de communication dédié à l'échange de fichiers informatique sur un réseau TCP/IP.



D . 1 . Déploiement de l'application sur le système embarqué cible.

Il faut exécuter le logiciel FileZilla, ouvrir une connexion FTP et transférer le programme **.tini** vers la carte TINI.

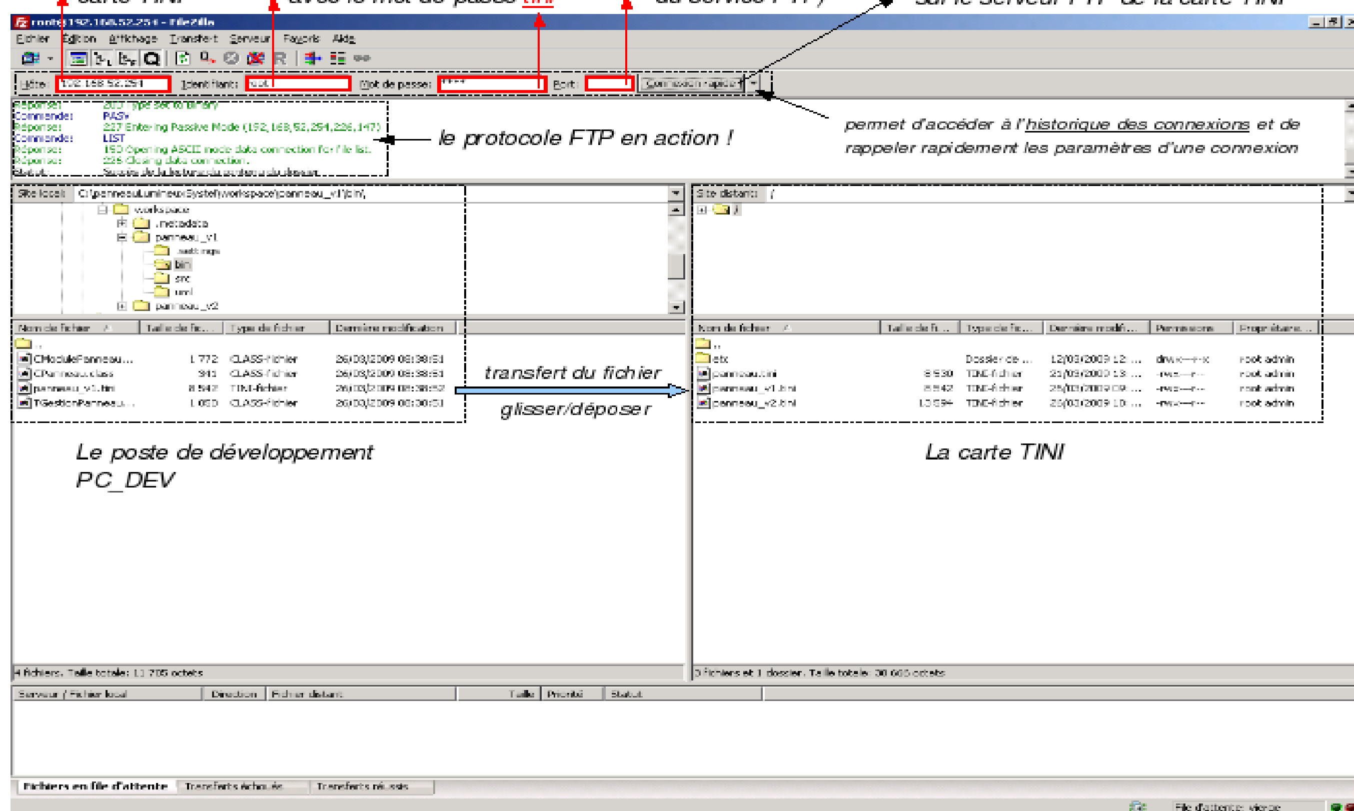
L'ouverture d'une session FTP sur la carte TINI nécessite une authentification (un compte identifié par un login et un mot de passe). Vous utiliserez l'identifiant suivant :

```
login : root

password : tini
```

Pour se connceter au serveur FTP de la carte TINI, vous pouvez utiliser le bouton **Connexion rapide** comme indiquer dans la capture d'écran fournie.

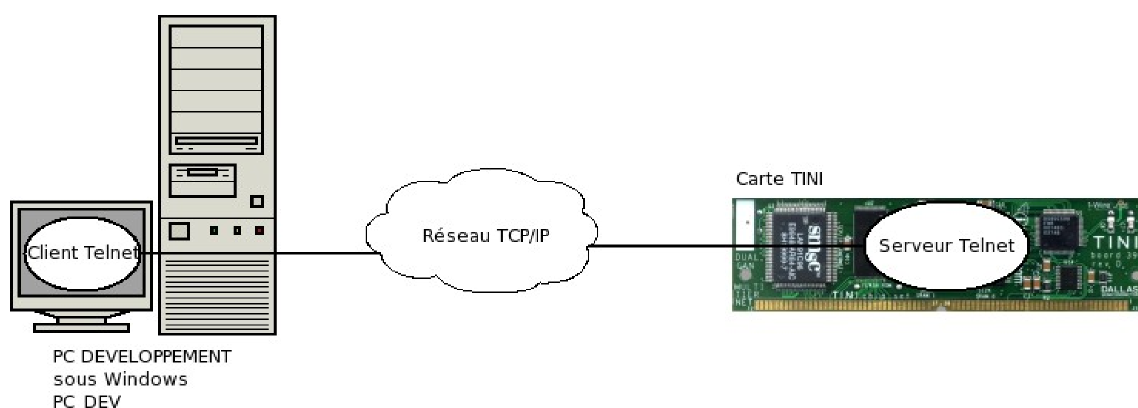
Une fois connecté, le transfert d'un fichier se fait par un simple glisser/déposer avec la souris.



Partie E : Exécution d'une application

E . 1 . Exécution de l'application.

La carte embarquée TINI ne disposant pas de terminal opérateur (clavier/écran), on va donc utiliser celui du poste de développement par l'intermédiaire d'un logiciel d'émulation de terminal via le réseau. Le protocole permettant l'émulation de terminal en réseau se nomme telnet. Le logiciel Putty va nous permettre d'ouvrir une session (accès à un terminal de connexion via le réseau sur la carte TINI).



L'ouverture d'une session sur la carte TINI nécessite une authentification (un compte identifié par un login et un mot de passe). Vous utiliserez l'identifiant suivant :

```
login : root

password : tini
```

E . 1 . 1 . Exécuter le logiciel Putty et ouvrir une session sur la carte TINI.

Les paramètres de connexion sont les suivants :

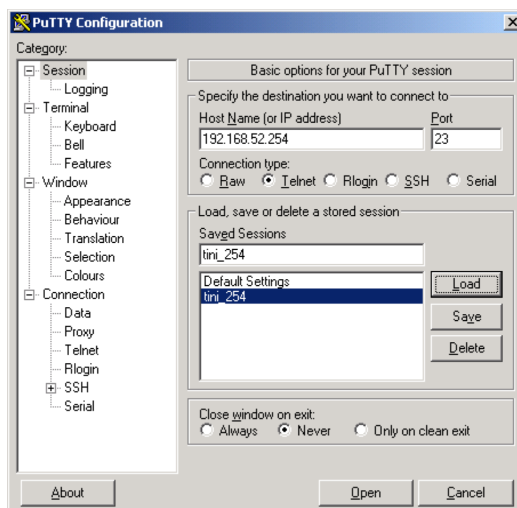
```
Host Name (or IP address) ← indiquer l'adresse IP de la carte TINI

Port ← 23 (le numéro de port du service Telnet)

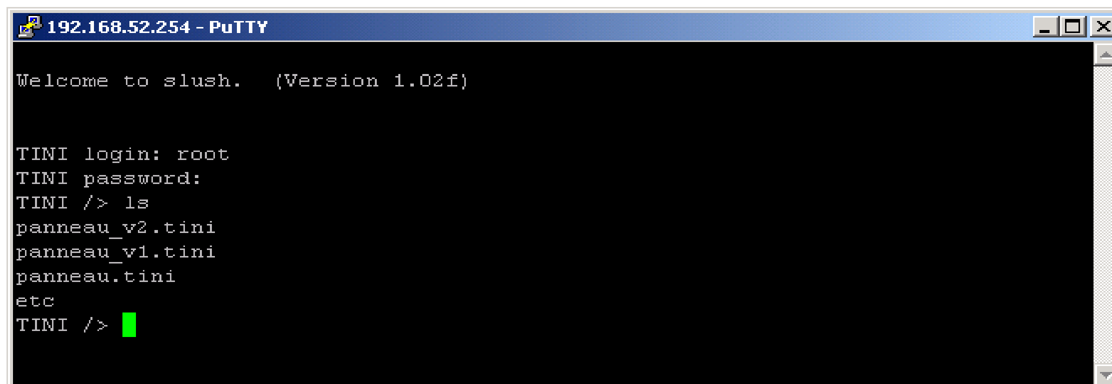
Connection type : Telnet
```

Ces paramètres peuvent être sauvegardés dans un profil de session avec le bouton Save et ré-utiliser plus tard avec le bouton Load.

Pour ouvrir une session, il faut ensuite cliquer sur le bouton Open !



Une fois connecté, il est possible d'exécuter des commandes sur la carte TINI. Ceci est possible car celle-ci dispose d'un système d'exploitation (TiniOs) et d'un interpréteur de commandes (nommé SLUSH).



Dans la session Telnet ouverte sur la carte TINI, lancer l'exécution du programme **.tini** avec la commande **java** (machine virtuelle), par exemple :

```
java panneau_v1.tini
```

```
192.168.52.254 - PuTTY
panneau_v1.tini
panneau.tini
etc
TINI /> java panneau_v1.tini
128
Clear
A 127
A 127
```

E . 2 . Arrêt d'une application qui ne répond plus.

Si l'application ne répond plus, il est possible d'arrêter le processus en cours de la manière suivante :

- Ouvrir une deuxième session telnet avec PuTTY
- Lister les processus, par exemple :

```
ps
```

- Tuer le processus désiré, par exemple :

```
kill 3
```

```
192.168.52.254 - PuTTY

Welcome to slush.


TINI login: root
TINI password:
TINI /> ps
3 processes
1: Java GC (Owner
2: init (Owner roo
3: panneau_v3.tini
TINI /> kill 3
TINI />
```